

Univerza v Ljubljani  
Fakulteta za *matematiko in fiziko*



# SEMINARSKA NALOGA

PROGRAMIRANJE 1

APLIKATIVNA MATEMATIKA

**Avtor naloge:**

MATIJA MATANIĆ

Ljubljana, 2023/24

	Ime naloge	Aktivna povezava	Število težavnostnih točk
1 del	Parent Gap	<a href="https://open.kattis.com/problems/parentgap">https://open.kattis.com/problems/parentgap</a>	2.5T
	Pizza Hawaii	<a href="https://open.kattis.com/problems/pizzahawaii">https://open.kattis.com/problems/pizzahawaii</a>	2.7T
	Race day	<a href="https://open.kattis.com/problems/raceday">https://open.kattis.com/problems/raceday</a>	3.7T
	Head Guard	<a href="https://open.kattis.com/problems/headguard">https://open.kattis.com/problems/headguard</a>	2.3T
	Hidden Password	<a href="https://open.kattis.com/problems/hidden">https://open.kattis.com/problems/hidden</a>	2.4T
	Spavanac	<a href="https://open.kattis.com/problems/spavanac">https://open.kattis.com/problems/spavanac</a>	1.4T
	Skupaj točk		15T

## Izjava

*Matija Matanić izjavljam, da sem seminarsko nalogo opravil samostojno in da sem njen avtor. To pomeni, da sem v poročilu točno označil vse tiste dele kode, ki sem jih povzel iz virov in vire navedel. Prav tako vem, da izjava o avtorstvu pomeni, da znam razložiti vsako podrobnost kode, ki jo oddajam. Zavedam se, da v primeru, če izjava prvega stavka ni resnična, kršim disciplinska pravila*

# **POROČILO 1. del**

# Parent Gap

Naloga je na spletni strani <https://open.kattis.com/problems/parentgap> in je vredna 2.5 točk.

## BESEDILO NALOGE

V Severni Ameriki in v mnogih drugih državah po svetu praznujejo materinski dan drugo nedeljo v maju, očetovski dan pa tretjo nedeljo v juniju. Razmik med materinskim in očetovskim dnem v katerem koli letu, ki se včasih imenuje tudi "parent gap" (starševski razmik), je vedno bodisi 5 tednov bodisi 6 tednov. Na primer, leta 2022 je bil materinski dan 8. maja, očetovski dan pa 19. junija, kar pomeni starševski razmik 6 tednov. Leta 2023 pa je bil materinski dan 14. maja, očetovski dan pa 18. junija, kar pomeni starševski razmik le 5 tednov.

Tvoja naloga je določiti starševski razmik za dano leto. Naslednje informacije ti bodo morda koristile. V letu, ki ni prestopno, imajo meseci naslednje število dni: januar: 31, februar: 28, marec: 31, april: 30, maj: 31, junij: 30, julij: 31, avgust: 31, september: 30, oktober: 31, november: 30, december: 31. V prestopnem letu ima februar en dodaten dan (29. februar), pravilo za prestopna leta pa je naslednje: leto, ki je deljivo s 4, je prestopno, razen če je deljivo s 100, pri čemer obstaja izjema od izjeme – če je leto deljivo s 400, je kljub temu prestopno leto.

### Vhodni podatki

Vhod je sestavljen iz enega samega leta,  $y$  ( $2000 \leq y \leq 3000$ ).

### Izhodni podatki

zhod naj bo starševski razmik za vneseno leto, bodisi "5 weeks" ali "6 weeks".

Primer vhodnih podatkov 1:

A dark-themed input field with a light gray border. Inside, the text "2022" is displayed in a light gray font. A small square icon with a diagonal line is visible in the top right corner of the field.

Primer izhodnih podatkov 1:

A dark-themed output field with a light gray border. Inside, the text "6 weeks" is displayed in a light gray font. A small square icon with a diagonal line is visible in the top right corner of the field.

Primer vhodnih podatkov 2:

A dark-themed input field with a light gray border. Inside, the text "2023" is displayed in a light gray font. A small square icon with a diagonal line is visible in the top right corner of the field.

Primer izhodnih podatkov 2:

A dark-themed output field with a light gray border. Inside, the text "5 weeks" is displayed in a light gray font. A small square icon with a diagonal line is visible in the top right corner of the field.

## OPIS PROBLEMA IN IDEJA REŠEVANJA

V tej nalogi sem moral izračunati, koliko tednov preteče med materinskim dnevom, ki je druga nedelja v maju, in očetovim dnevom, ki je tretja nedelja v juniju. Moram upoštevati koledar za običajno leto in za prestopno leto, saj se v prestopnem letu februar podaljša za en dan, kar lahko vpliva na to, na kateri dan v tednu padejo posamezni datumi. Na koncu sem ugotovil, ali je razlika 5 tednov ali 6 tednov.

Pri rešitvi sem najprej izračunal, kateri dan v tednu je 1. maj in 1. junij. Nato sem poiskal prvo nedeljo v mesecu in s tem določil, katera je druga oziroma tretja nedelja. Ko sem dobil točna datuma materinskega in očetovega dne, sem izračunal razliko med njima in jo preračunal v tedne. Tako sem dobil odgovor, ali je razmik 5 tednov ali 6 tednov.

## DOKAZ DA JE NALOGA REŠENA

[illegible]

# Pizza Hawaii

Naloga je na spletni strani <https://open.kattis.com/problems/pizzahawaii> in je vredna 2.7 točk.

## BESEDILO NALOGE

Potuješ v tuji državi. Čeprav si odprt za pokušanje lokalne hrane, preprosto ne moreš upreti skušnjavi, ko naletiš na italijansko restavracijo, ki ponuja pico. Na žalost je jedilni list napisan v tujem jeziku, zato so sezname sestavin pic zate nerazumljivi. Kaj boš storil?

Opaziš pa nekaj – vsaka pica ima italijansko ime, ki se ti zdi precej znano. Celo spomniš se, katere sestavine običajno vsebuje vsaka pica s tem imenom. To informacijo želiš uporabiti, da bi ugotovil možen pomen vsake besede na seznamu sestavin.

### Vhodni podatki

Prva vrstica vhoda podaja število testnih primerov  $t$  ( $0 < t \leq 20$ ). Prva vrstica vsakega vhoda podaja število pic  $n$  na meniju ( $1 \leq n \leq 60$ ). Naslednje  $3 \cdot n$  vrstic opisuje pice na meniju. Opis vsake pice se začne z eno vrstico, ki vsebuje ime pice. Ime pice je sestavljeno iz med 3 in 20 velikih in malih črk. Naslednja vrstica se začne s celo številko  $m_i$ , ki podaja število sestavin te pice na  $1 \leq m_i \leq 20$ . Preostanek vrstice vsebuje  $m_i$  sestavin, ločenih s presledki. Vsaka sestavina je beseda, sestavljena iz med 2 in 20 malih črk. Tretja vrstica vsakega opisa pice podaja sestavine v tvojem maternem jeziku v enakem formatu. Upoštevaj, da se število sestavin lahko razlikuje, saj vsaka restavracija lahko uporablja nekoliko drugačne sestavine za pice z istim imenom, zato se sestavine, ki si jih zapomniš za določeno pico, morda ne bodo ujemale s pravimi sestavinami.

### Izhodni podatki

Za vsak testni primer izpiši vse pare besed  $(w_1, w_2)$ , kjer je  $w_1$  sestavina v tujem jeziku, ki bi lahko bila enaka sestavina kot  $w_2$ , saj se  $w_1$  in  $w_2$  pojavljata pri istem naboru pic. Pare uredi v naraščajočem leksikografskem vrstnem redu glede na  $w_1$ , v primeru enakosti pa v naraščajočem leksikografskem vrstnem redu glede na  $w_2$ . Med različnimi testnimi primeri izpiši prazno vrstico.

Primer vhodnih podatkov 1:

```
2
3
Hawaii
4 tomaten schinken ananas kaese
4 pineapple tomatoes ham cheese
QuattroStagioni
6 tomaten kaese salami thunfisch
6 mushrooms tomatoes cheese pepperoni
Capricciosa
6 champignons kaese tomaten artichoke
5 cheese tomatoes mushrooms ham
1
Funghi
3 tomaten kaese champignons
3 cheese tomatoes mushrooms
```

Primer izhodnih podatkov 1:

```
(ananas, pineapple)
(artischocken, artichoke)
(champignons, mushrooms)
(kaese, cheese)
(kaese, ham)
(kaese, tomatoes)
(oliven, artichoke)
(salami, peppers)
(salami, salami)
(spinat, peppers)
(spinat, salami)
(thunfisch, peppers)
(thunfisch, salami)
(tomaten, cheese)
(tomaten, ham)
(tomaten, tomatoes)

(champignons, cheese)
(champignons, mushrooms)
(champignons, tomatoes)
(kaese, cheese)
(kaese, mushrooms)
(kaese, tomatoes)
(tomaten, cheese)
(tomaten, mushrooms)
(tomaten, tomatoes)
```

## OPIS PROBLEMA IN IDEJA REŠEVANJA

V tej nalogi imamo za vsako pico dva seznama sestavin: enega v tujem jeziku in enega v domačem. Iz teh podatkov želimo odkriti morebitne pare besed, ki se ujemajo v obeh jezikih, tako da pogledamo, katere sestavine se hkrati pojavljajo pri enaki pici. Na koncu te pare izpišemo v urejenem (leksikografskem) vrstnem redu.

Za rešitev sem se lotil shranjevanja možnih prevodov na ta način: za vsako tujo besedo sem hranil množico vseh domačih besed, ki bi lahko ustrezale, in obratno tudi za vsako domačo besedo množico vseh možnih tujih besed. Pri vsaki pici sem moral posodobiti te množice tako, da sem obdržal le še tiste besede, ki se pojavljajo sočasno. Na koncu sem le še preveril, ali se možnost prevajanja ujema v obe smeri (tuj→dom in dom→tuj), in tiste pare sem izpisal urejene po abecedi. S tem sem izključil besede, ki sicer nakazujejo povezavo, a se ne ujemajo v vseh primerih.

## DOKAZ DA JE NALOGA REŠENA

DATE	PROBLEM	JUDGEMENT	RUNTIME	LANGUAGE	TEST CASES
16:43:19 ✔ ✔ ✔ ✔	Pizza Hawaii	✔ Accepted	0.09 s	Python 3	4/4

# Race Day

Naloga je na spletni strani <https://open.kattis.com/problems/raceday> in je vredna 3.7 točk.

## BESEDILO NALOGE

Ti si odgovoren za spremljanje in poročanje rezultatov tekačev v tekmi. Želiš razviti programski paket, ki ti bo to nalogo olajšal. Tvoj program naj prikaže rezultate vseh tekačev v eni tabeli, pri čemer bo vsakega tekača razvrstil glede na njegov skupni čas in vmesne čase. Vmesni časi se merijo od začetka tekme do določenih točk (npr. v teku na 5 km so lahko vmesne točke na 2 km in 4 km). Lahko predpostaviš, da vsaka tekma vedno vsebuje 2 vmesni točki ter končni čas. Celotna tabela mora biti urejena po tekačevem priimku in nato po imenu. Vsi tekači imajo unikatna imena.

Ena od težav, s katero se mora tvoj program spopasti, je zbiranje podatkov iz različnih virov. Ker več oseb beleži čase tekačev na posameznih točkah merjenja, lahko informacije prispejo v katerem koli vrstnem redu in niso združene po tekačih. Vsak izmerjeni čas je označen s tekačevo štartno številko in lokacijo, s katere je bil čas zabeležen. Tvoj program mora zbrati vse podatke po tekačih in jih organizirati na zgoraj opisan način.

### Vhodni podatki

Vhod vsebuje podatke iz največ 10 tekem. Vsako tekmo obdelaj ločeno. Vhod za posamezno tekmo se začne s pozitivnim celoštevilskim številom  $1 \leq n \leq 500$ , ki mu sledi seznam  $n$  opisov tekačev. Opis vsakega tekača vsebuje tekačevo ime (ime sledi priimku) in štartno številko. Štartna številka je 5-mestno število (lahko vsebuje vodilne ničle) in je unikatna za tega tekača v tej tekmi. Za opisi tekačev sledi  $3n$  zapisov merjenja časa. Vsak zapis vsebuje tri dele: štartno številko, lokacijo merjenja časa (S1 za prvo vmesno točko, S2 za drugo vmesno točko ali F za cilj) ter izmerjeni čas v formatu MM:SS. Vsa imena so sestavljena le iz črkovnih znakov (a–z).

Zadnjem testnem primeru sledi vrstica, ki vsebuje samo številko 0.

### Izhodni podatki

Za vsako tekmo izpiši formatirano tabelo, ki vsebuje priimek, vejico, ime, štartno številko, nato čas in uvrstitev za obe vmesni točki ter cilj. Stolpec z imenom naj bo širok 20 znakov in levo poravnan. Vsi preostali stolpci naj bodo široki 10 znakov in desno poravnani. Vsaka tabela mora imeti glavo, formatirano tako, kot je prikazano v vzorčnem izpisu.

Če imata dva ali več tekačev enak čas, morajo imeti enako uvrstitev kot najvišje možno mesto, pri čemer se rangiranje za počasnejše tekače ne sme spreminjati. Na primer, če si dva tekača delita prvo mesto, oba prejmeta uvrstitev 1, naslednji tekač pa dobi uvrstitev 3.

Sledi formatu iz vzorčnega izpisa.



Primer vhodnih podatkov 1:

```
5
Bert Trista 00227
Joanne Valeria 00248
Elaina Eduardo 00615
Iona Scott 00234
Jon Idella 00014
00234 S2 31:05
00234 F 55:25
00248 S2 37:24
00615 S1 17:54
00227 F 52:41
00615 S2 32:22
00234 S1 16:58
00248 F 53:18
00014 F 56:26
00014 S1 17:57
00227 S1 20:40
00248 S1 21:13
00014 S2 32:22
00227 S2 34:55
00615 F 51:22
3
Bernardo Lesa 31183
Thad Kellie 28712
Claudia Brigitte 38522
38522 S1 25:21
38522 S2 30:36
31183 S1 23:15
31183 S2 29:28
28712 F 53:21
28712 S2 33:31
31183 F 41:17
28712 S1 23:50
38522 F 45:09
0
```

Primer izhodnih podatkov 1:

NAME	BIB	SPLIT1	RANK	SPLIT2	RANK	FINISH	RANK
Eduardo, Elaina	00615	17:54	2	32:22	2	51:22	1
Idella, Jon	00014	17:57	3	32:22	2	56:26	5
Scott, Iona	00234	16:58	1	31:05	1	55:25	4
Trista, Bert	00227	20:40	4	34:55	4	52:41	2
Valeria, Joanne	00248	21:13	5	37:24	5	53:18	3

NAME	BIB	SPLIT1	RANK	SPLIT2	RANK	FINISH	RANK
Brigitte, Claudia	38522	25:21	3	30:36	2	45:09	2
Kellie, Thad	28712	23:50	2	33:31	3	53:21	3
Les, Bernardo	31183	23:15	1	29:28	1	41:17	1

## OPIS PROBLEMA IN IDEJA REŠEVANJA

V tej nalogi je treba obdelati podatke o tekačih v eni ali več tekmah, kjer vsak tekač nastopa z imenom, priimkom in štartno številko, hkrati pa je pri vsakem tekaču zabeležen čas na dveh vmesnih točkah in na cilju. Zbrati je treba vse podatke in jih združiti v skupno tabelo, ki je urejena po priimku in imenu tekača. V tabeli je poleg imen in števil prikazan čas na vmesnih točkah in na cilju, pri čemer je treba izračunati tudi uvrstitve tako, da se tekači z enakim časom uvrstijo na isto mesto, naslednji rezultat pa ohrani ustrezno zaporedno številko.

Do rešitve sem prišel tako, da sem najprej ustvaril strukturo, ki je hranila podatke o posameznem tekaču (ime, priimek, štartna številka in trije časi). Nato sem vse tekače razvrstil trikrat: najprej glede na čas na prvi vmesni točki, potem na drugi, in nazadnje na cilju. Pri vsaki razvrstitvi sem določil uvrstitev tako, da sem za čas, ki je enak prejšnjemu, pripisal isto mesto, drugače pa sem novega tekmovalca uvrstil na naslednje mesto v vrsti. Na koncu sem tekmovalce še enkrat razvrstil po priimku in imenu ter pripravil formatiran izpis, ki ustreza zahtevam glede širine in poravnave stolpcev.

## DOKAZ DA JE NALOGA REŠENA

DATE	PROBLEM	JUDGEMENT	RUNTIME	LANGUAGE	TEST CASES
17:05:18 ✓ ✓	Race Day	✓ Accepted	0.10 s	Python 3	2/2

# Head Guard

Naloga je na spletni strani <https://open.kattis.com/problems/headguard> in je vredna 2.3 točk.

## BESEDILO NALOGE

Ti si glavni stražar v Kraljevem gradu Modre gore.

Tvoja naloga je preprosta: voditi evidenco o tem, koliko vrst bitij je vstopilo skozi katera vrata in v kakšnem vrstnem redu.

Grad je znan po številnih vratih, vendar so ta včasih lahko odprta ali zaprta.

### Vhodni podatki

Vhod je sestavljen iz 1 do 15 vrstic. Vsaka vrstica vsebuje niz, sestavljen iz 1 do 500 malih črk angleške abecede.

### Izhodni podatki

V vsaki vrstici izpiši število zaporednih znakov, ki so se pojavili, pri čemer naj bo število takoj pred znakom. Na primer, niz "aaaabb" postane "4a2b". Če se isti znak pojavi dvakrat v različnih zaporedjih, kot v nizu "aaabbaaa", ga ne smeš združiti v "6a2b". Ker je vrstni red pomemben za stražarje, mora ostati v obliki "3a2b3a".

Primer vhodnih podatkov 1:

```
aaaaabbccddeee
kklllmmnnnnn
aaaabb
aaabbaaa
aaabbzzzzzoioussslllwwkppddmmnnnnnnnnnnnnnnppppqq
```

Primer izhodnih podatkov 1:

```
5a3b2c2d3e
2k3l2m5n
4a2b
3a2b3a
3a2b6z1o1i1o1u4s3l3w1k2p3d1m15n4p2q
```

## OPIS PROBLEMA IN IDEJA REŠEVANJA

V tej nalogi je treba obdelati do 15 vnosov, pri čemer vsak vnos sestavlja niz z do 500 malimi črkami angleške abecede. Zahteva je, da se za vsak niz izpiše zaporedje znakov skupaj s številom ponovitev, pri čemer se za vsako zaporedje istih znakov izpiše število, ki mu neposredno sledi znak (na primer "aaaabb" postane "4a2b"). Ločenih zaporedij enakega znaka se ne sme združevati nazaj, saj vrstni red in razporeditev igrata ključno vlogo.

V svoji rešitvi sem vsak vnos prebral v zanko, nato pa v ločeni funkciji preveril zaporedne znake. Ko sem naletel na nov znak, sem staro zaporedje zaključil, v izhod shranil število in znak zaporedja, nato pa začel šteti novo. Zadnji znak sem dodal po koncu glavne zanke. Tako sem poskrbel, da je bil vsak

zaporedni sklop zapisan v obliki "<število><znak>", in da so bile morebitne ponovitve prenešene v izpis takoj, ko se je znak spremenil.

## DOKAZ DA JE NALOGA REŠENA

[illegible]

# Hidden Password

Naloga je na spletni strani <https://open.kattis.com/problems/hidden> in je vredna 2.4 točk.

## BESEDILO NALOGE

Insecure Inc. se je po neuspelem poskusu razvoja novega šifrirnega standarda odločil za spremembo smeri. Njihov novi projekt je sistem gesel, pri katerem je geslo skrito znotraj drugega niza znakov, ki ga imenujemo sporočilo. Vendar je pomembno, da ima sporočilo določeno lastnost glede na skrito geslo.

Predpostavimo, da geslo sestavljajo znaki  $c_1 c_2 \dots c_p$  (ti znaki se ne nujno razlikujejo). Da bi bilo sporočilo veljavno za dano geslo, mora ob pregledu sporočila v določenem zaporedju veljati naslednje pravilo, če začnemo na začetku sporočila in poiščemo kateri koli znak iz množice  $\{c_1, \dots, c_p\}$ , mora biti prvi najdeni znak prav  $c_1$ . Nato nadaljujemo iskanje naprej od te točke in poiščemo naslednji znak iz množice  $\{c_2, \dots, c_p\}$ , pri čemer mora biti naslednji najdeni znak prav  $c_2$ . Po enakem principu mora  $c_3$  biti naslednji znak  $\{c_3, \dots, c_p\}$ , in tako naprej, dokler ne dosežemo znaka  $c_p$ .

Primer, če je geslo ABC, je niz HAPPYBIRTHDAYCACEY veljavno sporočilo.

- Prvi najdeni znak iz množice {A, B, C} je A (začetni H ni pomemben).
- Po A je naslednji najdeni znak iz množice {B, C} prav B.
- Po B je naslednji najdeni znak iz množice {C} prav C.

Dodatni A in C v CACEY niso pomembni, saj smo že zaključili geslo s prvim C.

Če je geslo ABC, niz TRAGICBIRTHDAYCACEY ni veljavno sporočilo.

- A je prvi znak iz množice {A, B, C}. Naslednji najdeni znak iz množice {B, C} je C namesto B, kar krši pravilo.

Niz HAPPYBIRTHDAY ni veljavno sporočilo za geslo ABC, saj C sploh ne nastopi.

Če ima geslo ponavljajoče se črke, kot na primer SECRET, je niz SOMECHORESARETOUGH veljavno sporočilo. V nasprotju s tem niz SOMECHEERSARETHOUGH ni veljavno sporočilo, ker se pojavi dodatni E v trenutku, ko se pričakuje R.

### Vhodni podatki

Vhod je sestavljen iz ene vrstice, ki vsebuje dva niza. Prvi niz je geslo dolžine P, pri čemer velja  $3 \leq P \leq 8$ . Drugi niz ima dolžino S, pri čemer velja  $10 \leq S \leq 40$ . Oba niza vsebujeta izključno velike črke. Noben niz ne sme vsebovati presledkov, malih črk, števil ali drugih posebnih znakov.

### Izhodni podatki

Izpiši eno vrstico z besedo PASS, če je drugi niz veljavno sporočilo za geslo, sicer izpiši FAIL.

Primer vhodnih podatkov 1:

```
ABC HAPPYBIRTHDAYCACEY
```

Primer izhodnih podatkov 1:

```
PASS
```

Primer vhodnih podatkov 2:

```
ABC TRAGICBIRTHDAYCACEY
```

Primer izhodnih podatkov 2:

```
FAIL
```

Primer vhodnih podatkov 3:

```
ABC HAPPYBIRTHDAY
```

Primer izhodnih podatkov 3:

```
FAIL
```

Primer vhodnih podatkov 4:

```
SECRET SOMECHORESARETOUGH
```

Primer izhodnih podatkov 4:

```
PASS
```

Primer vhodnih podatkov 5:

```
SECRET SOMECHEERSARETOUGH
```

Primer izhodnih podatkov 5:

```
FAIL
```

## OPIS PROBLEMA IN IDEJA REŠEVANJA

V tej nalogi je podan niz, ki predstavlja geslo (npr. »ABC«), in drug niz, ki predstavlja sporočilo. Preveriti je treba, ali se geslo skriva v sporočilu na točno določen način: znaki gesla se morajo pojaviti zaporedno, pri tem pa ne sme biti preskočen noben znak gesla ali pa najden kateri od kasnejših znakov gesla prezgodaj. Tako mora, na primer, za prvo črko gesla najprej nastopiti prva ustrezna črka v sporočilu, za drugo črko gesla šele naslednja ustrezna črka itd. Če do konca sporočila ni mogoče pravilno slediti vsem znakom gesla, velja, da gesla ni mogoče najti.

V svoji rešitvi sem se odločil, da skozi sporočilo potujem z dvema kazalcema: eden spremlja trenutno mesto v sporočilu, drugi pa trenutni znak, ki ga iščem v geslu. Vsakič ko najdem ustrezen znak, kazalec v geslu povečam. Če pa v sporočilu najdem znak, ki se pojavi kasneje v geslu, še preden pridem do njega, takoj označim, da geslo ne ustreza. Če mi uspe prebrati vse znake v geslu, preden mi zmanjka sporočila, izpišem 'PASS', sicer 'FAIL'.

## DOKAZ DA JE NALOGA REŠENA

DATE	PROBLEM	JUDGEMENT	RUNTIME	LANGUAGE	TEST CASES
17/12/12	Hidden Password	Accepted	0.04 s	Python 3	32/32

# Spavanac

Naloga je na spletni strani <https://open.kattis.com/problems/spavanac> in je vredna 1.4 točk.

## BESEDILO NALOGE

Vsako šolsko jutro Mirka zbudi zvok budilke. Ker je nekoliko pozabljiv, pogosto pusti budilko vklopljeno tudi v soboto zjutraj. To ni preveč slabo, saj se počuti odlično, ko ugotovi, da mu ni treba vstati iz svoje tople in udobne postelje.

To mu je tako všeč, da bi rad to doživel tudi druge dni v tednu. Njegov prijatelj Slavko mu je predlagal preprosto rešitev: naj si budilko nastavi 45 minut prej, da bo lahko vsak dan užival v udobju postelje, popolnoma buden, še 45 minut.

Mirko se je odločil upoštevati njegov nasvet, vendar njegova budilka uporablja 24-urni zapis časa, kar mu povzroča težave pri nastavitvi ure. Pomagaj Mirku in napiši program, ki bo sprejel časovni žig v 24-urnem zapisu in izpisal nov časovni žig, ki je 45 minut prej, prav tako v 24-urnem zapisu.

Če sam nisi seznanjen s 24-urnim zapisom časa, naj te spomnimo, da se začne ob 0:00 (polnoč) in konča ob 23:59 (ena minuta pred polnočjo).

### Vhodni podatki

Prva in edina vrstica vhoda bo vsebovala natanko dve celi števili  $H$  in  $M$  ( $0 \leq h \leq 23, 0 \leq M \leq 59$ ), ločeni z enim presledkom. Vhodni čas je podan v 24-urnem zapisu, kjer  $H$  označuje ure,  $M$  pa minute.

### Izhodni podatki

Prva in edina vrstica izhoda naj vsebuje natanko dve celi števili, čas 45 minut pred vhodnim časom.

Primer vhodnih podatkov 1:

```
10 10
```

Primer izhodnih podatkov 1:

```
9 25
```

Primer vhodnih podatkov 2:

```
0 30
```

Primer izhodnih podatkov 2:

```
23 45
```

Primer vhodnih podatkov 3:

```
23 40
```

Primer izhodnih podatkov 3:

```
23 40
```



## OPIS PROBLEMA IN IDEJA REŠEVANJA

V tej nalogi je treba vhodni čas, zapisan v 24-urni obliki (podani sta ura in minuta), premakniti 45 minut nazaj in rezultat izpisati v enaki 24-urni obliki. Če ob premiku pridemo do negativnih minut, to pomeni, da moramo iti na prejšnji dan, pri čemer se ura ustrezno prilagodi tako, da ostane v okviru 24-urnega formata.

Pri svoji rešitvi sem najprej pretvoril vhodno uro in minuto v skupno število minut od začetka dneva ter odštel 45. Če je bil novi rezultat manjši od nič, sem temu znesku prištel 1440 (število minut v enem dnevu), s čimer sem se izognil težavam z negativnimi vrednostmi. Na koncu sem s celoštevilskim deljenjem in ostankom pri deljenju ponovno izračunal novo uro in nove minute ter ju izpisal.

## DOKAZ DA JE NALOGA REŠENA

DATE	PROBLEM	JUDGEMENT	RUNTIME	LANGUAGE	TEST CASES
17:13:34	Spavanje	✓ Accepted	0.04 s	Python 3	13/13
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓					